ภาษิต ภัทรมูล

ลีก ๆ กับ Visual FoxPro 5.0 ตอนที่ 4

การทำกราฟด้วย Visual FoxPro โดยการใช้ OLE Bound Control

กลับมาตามสัญญาครับ ที่เคยเกริ่นในตอนที่แล้วว่าจะพูดถึงเรื่องการทำกราฟใน Visual FoxPro ซึ่งมีความยุ่งยากพอสมควร และอีกส่วนหนึ่งที่จะเสริมคือการใช้ DDE (Data Dynamic Exchange) สำหรับการแลกเปลี่ยนข้อมูลระหว่างแอพพลิเคชันในวินโดวส์

ผู้เขียนพบกับความประหลาดใจว่า การทำกราฟใน Visual FoxPro โดยใช้ Graph Wizard นั้น เมื่อใช้ร่วมกับ MSOffice 97 ปรากฏว่าใช้ไม่ได้ โปรแกรมจะฟ้องขึ้นมาทุกครั้งว่า หา MSGraph ไม่พบ หรือไม่ก็รันไม่สำเร็จ แต่เหตุการณ์นี้บางเครื่องก็ไม่ปรากฏ สามารถใช้ MSGraph ได้ตามปกติ แต่เมื่อเขียนเป็นโค้ดก็ยังต้องมาทำกราฟใหม่ทุกครั้ง

เราคงไม่กล้าเสี่ยงกับความไม่แน่นอนเช่นนี้ จะมีวิธีไหนที่จะสามารถรัน MSGraph ได้ทุก ๆ เครื่อง และแสดงผลออกมาโดยไม่ต้องกลับมาทำใหม่ ซึ่งถ้าเป็นเช่นนี้แสดงว่า การแสดงข้อมูลเป็นกราฟทุกครั้งนั้น ยูสเซอร์จะต้องมานั่งคลิกเลือก Table, แกน X, แกน Y หรือระบุ Label เองอย่างนั้นหรือ?

หลังจากการงมโข่งลองผิดลองถูกเป็นเวลาหลายวัน จึงพอจะพบแสงสว่างรำไรในการทำกราฟด้วย Visual FoxPro

เนื้อหา

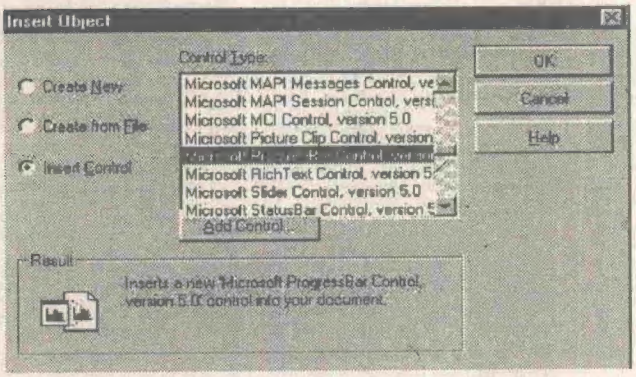
นำเสนอการทำกราฟด้วย Visual FoxPro ใน 2 แนวทางด้วยกัน

1. ใช้ OLE Bound Control
2. ใช้ DDE ติดต่อกับ Ms.Excel

การทำกราฟโดย OLE Bound Control

OLE (Object Linking and Embedding) คืออะไร

การเชื่อมโยงและฝังในวัตถุเป็นคุณสมบัติที่วินโดวส์มี นั่นคือ การที่เราสามารถถ่ายโอนและใช้ข้อมูลร่วมกันท่ามกลางโปรแกรมใช้งานบนวินโดวส์ โดยไม่ต้องสำเนาหรือตัดแปะวัตถุเพื่อแลกเปลี่ยนข้อมูลเลย หมายถึงเราไม่จำเป็นต้องออกจากโปรแกรมที่ใช้งานอยู่ แต่สามารถเข้าไปใช้งานโปรแกรมข้างนอกได้



รูปที่ 1

ในวินโดวส์จะมีโปรแกรมที่ใช้ OLE อยู่ 2 ประเภท คือ

1. ตัวขอรับบริการ (Client) คือตัวโปรแกรมที่สามารถใช้การเชื่อมโยงและการฝังในวัตถุในโปรแกรมได้

2. ตัวบริการ (Sever) คือโปรแกรมใช้งานที่สามารถเรียกใช้ได้จากโปรแกรมอื่น ๆ

บทบาทของ Visual FoxPro จะเป็น Client

การใช้ OLE ใน Visual FoxPro

จะมีการใช้งานโดยผ่านใน 2 วิธี คือ

1. การใช้คำสั่ง VERB โดยมีรูปแบบไวยากรณ์

```
@ 0,0 say ชื่อวัตถุ OLE VERB "คำสั่ง"
```

เช่น เราต้องการเล่นเสียงที่บันทึกอยู่ใน Table ของ FoxPro เราก็สามารถใช้คำสั่งดังนี้

```
@0,0 say x_sound VERB "Play"
```

อย่างไรก็ตามใน Visual FoxPro ได้เพิ่ม OLE Control และฟังก์ชันที่เกี่ยวข้องกับ OLE ออกหลายตัวให้เราเรียกใช้ได้ง่ายกว่านี้ สำหรับ Command ที่อยู่หลังคำว่า VERB เราสามารถเปิดดูว่าแอพพลิเคชันตัวไหนสามารถใช้ได้ใน Registry Editor > My computer> HKEY_CLASSES_ROOT ช่วงล่าง ๆ จะมีรายชื่อแอพพลิเคชันเปิดเข้าไปในแต่ละตัว หาใน Folder ที่ชื่อว่า VERB

2. การใช้ OLE Control

2.1 OLE Container Control เป็น Control ที่สามารถใช้ OLE Object ในรูปแบบ Control เช่น ActiveX หรือ Control ที่อยู่ใน Library นามสกุล .OCX หรือ Object อื่น ๆ เช่น รูปภาพ ซึ่งต้องการให้ฝังใน Form โดยไม่ต้องการปรับแต่งคุณสมบัติภายใน

เมื่อค้นคว้ามาถึงตอนนี้จึงพบวิธีแก้ปัญหาที่คาใจมานาน มีของดีต้องเอามาฝากครับ นั่นคือวิธีใช้ Control ActiveX ซึ่งจะขอกล่าวถึงพอเป็นสังเขป ดูรูปที่ 1

ขั้นแรกให้เราคลิกที่ OLE Container Control หน้าจอจะเข้าสู่ Windows insert object ให้เราเลือก Insert Control แล้วเลื่อนหา Microsoft ProgressBar Control เท่านี้เราก็จะสามารถใช้ ActiveX ได้ตามใจปรารถนา ความจริงมันง่ายนิดเดียวแต่เราหาไม่เห็นตรงส่วนนี้ ทั้ง ๆ ที่ตะลุยอ่าน Help อย่างหนักในช่วงที่เขียน ProgressBar ขึ้นใช้เอง

สำหรับวิธีใช้ก็คล้าย ๆ กับ ProgressBar ตัวที่สร้างขึ้นเองนั่นแหละ คือเราต้องเข้าไปเซ็ตในส่วนของคุณสมบัติความก้าวหน้าเท่านั้นโค้ดยาวเหมือนเดิมครับ...อย่าเพิ่งดีใจ

2.2 OLE Bound Control ซึ่งมีคุณสมบัติพิเศษคือ เราสามารถดับเบิลคลิกเพื่อเข้าไปแก้ไขวัตถุได้โดยตรง และยังมีคุณสมบัติที่เป็นของตนเอง สามารถสั่งการได้เสมือนเป็นโปรแกรมอีกโปรแกรมหนึ่ง เราสามารถเรียกใช้โดยตั้งค่าผ่านฟิลด์ประเภท General

ในฉบับนี้จะขอเน้น OLE Bound Control ที่เราจะเอามาทำกราฟครับ

เมื่อเราจะทำกราฟ

อันดับแรกคือ สร้างแฟ้มข้อมูลขึ้นมา 1 แฟ้ม มีโครงสร้างดังรูปที่ 2

สมมติให้แฟ้มนี้มีไว้สำหรับเก็บข้อมูลจำนวนต้นไม้ในสนามหลวงช่วงปี 2540-2541 มีข้อมูลในแฟ้มดังนี้ (ดูรูปที่ 3)

เราต้องการจะนำข้อมูลมานำเสนอในรูปของกราฟแท่งแนวนอนเพื่อแสดงการเพิ่มหรือลดของจำนวนต้นไม้ประเภทต่าง ๆ ในแต่ละปี

ลงไม้ลงมือ

สร้างฟอร์มขึ้นหนึ่งฟอร์ม โดยให้มี OLE Bound Control อยู่ในฟอร์มหนึ่งตัว ขยายขนาดตามชอบครับ (รูปที่ 4)

ก่อนที่เราจะมาเขียนโค้ด เราควรมาทำความรู้จักกับโปรแกรมกราฟที่เราจะดึงมันมาใช้ใน OLE Bound Control เสียก่อน

Visual FoxPro รู้จักว่า Graph ตัวนี้เป็น Class ตัวหนึ่งที่มีชื่อว่า msgraph.chart ซึ่งก็มีคุณสมบัติเป็นของตัวเองหลายอย่าง ไม่ว่าจะเป็น Title, Data, legend, bartype เป็นต้น

การที่จะแสดงข้อมูลออกมาเป็นกราฟได้ ส่วนที่สำคัญที่ MSGraph จะนำไปประมวลผลคือ รูปแบบของ Data ที่เราจะต้องใส่เข้าไปให้ถูกต้องมีรูปแบบดังนี้ครับ (สมมติว่าเราจะใช้แฟ้มข้อมูลต้นไม้มาเป็นตัว Presenter)

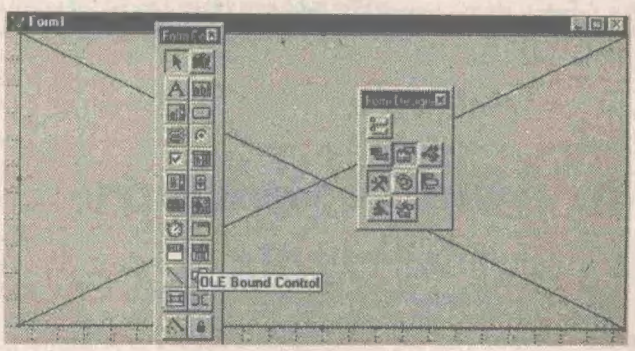
```
"ต้นมะพร้าว"+chr(9)+"ต้นมะขาม"+chr(9)+
"ต้นตาล"+chr(9)+"อื่น ๆ "; &&หัวเรื่อง (Legend)
+chr(13)+chr(10)+"2540"+chr(9)+"2000"+
chr(9)+"5000"+ chr(9)+"1500"+ chr(9)+
chr(9)+"1100" +chr(13)+chr(10);  && Series ที่ 1
(ข้อมูล พ.ศ.2540)
+"2541"+chr(9)+"2154"+ chr(9)+"6000"+
chr(9)+"2000"+ chr(9)+ "1400" +chr(13)+chr(10)
&& Series ที่ 1 (ข้อมูล พ.ศ.2541)
```

	Name	Type	Width
↕	namet	Character	10
	y1997	Numeric	10
	y1998	Numeric	10

รูปที่ 2

Namet	Y1997	Y1998
มะพร้าว	2000	2154
มะขาม	5000	6000
ตาล	1500	2000
อื่นๆ	1100	1400

รูปที่ 3



รูปที่ 4

ต้องพยายามทำความเข้าใจกับมันสักหน่อย เพราะถ้าเราสับสนกับโครงร่างเสียแล้วการเขียนโค้ดจัดการกับมันย่อมไม่สามารถทำได้ ดูในภาพกว้างคือมันจะต้องการส่วนหัวเรื่องจนครบ (ในที่นี้คือชื่อต้นไม้ในฟิลด์ Namet) ต่อจากนั้นคือเรื่องของ Series ในที่นี้คือชื่อปี ต่อไปจึงตามด้วยข้อมูลซึ่งมีความสัมพันธ์กับหัวเรื่อง ต้องใส่ตรงกันข้อมูลจึงจะออกมาถูกต้อง

แต่ละส่วนของข้อมูลจะคั่นด้วย Chr(9) ซึ่งก็คือ TAB แต่ละ Legend กับ Series จะคั่นกันด้วย Chr(13)+Chr(10) แสดงถึงการจบในแต่ละส่วน

ส่วนของ Data นี้เราต้องเขียนโค้ดจัดการกับมัน ด้วยเหตุจำเป็น 2 ประการ

1. ข้อมูลที่เราประมวลผลต้องมีการเปลี่ยนแปลง ดังนั้นตัวแปรต่าง ๆ ที่ปรากฏใน Data จึงไม่สามารถใส่ค่าลงไปตรง ๆ ในโค้ดได้

2. คงไม่มีใครมานั่งประดิษฐ์ประดอยตรงนั้นนิดตรงนี้หน่อยโดยที่ได้งานเล็กน้อย ถ้าเราทำเป็นโมดูลเก็บไว้ก็สะดวกในการนำมาแก้ไขปรับปรุงในภายหลัง หรือใช้ในงานอื่น ๆ ได้สะดวก

Property ที่น่าสนใจ

1. Hastitle ต้องระบุส่วนนี้ให้มีค่าเป็น .T. เราจึงจะกำหนด Title ของกราฟได้

2. Haslegend ต้องระบุส่วนนี้ให้มีค่าเป็น .T. Legend ของกราฟจึงปรากฏ

3. ChartTitle.Caption ใช้ระบุ Caption ของ Title

4. Autoformat (graphtype,1) ใช้ระบุชนิดของกราฟที่จะแสดงโดยรับค่าเป็นรหัสตัวเลข

เช่น

กราฟแท่งแนวนอน	สามมิติ	-4099
กราฟแท่งแนวตั้ง	สามมิติ	-4100
กราฟวงกลม	สามมิติ	-4102

ฟิลด์ชนิด General(g)

เราจะใช้ฟิลด์ชนิด General ในการเก็บข้อมูลประเภท OLE เช่น Speadsheet ของ Excel แฟ้มของ Word รูปภาพ ตลอดจนแฟ้มซึ่งสร้างจากโปรแกรมอื่น ๆ สามารถเก็บมีขนาดใหญ่มากจำกัดตามเนื้อที่ว่างใน HardDisk

คำสั่งที่เกี่ยวข้องคือ

1. APPEND GENERAL ชื่อฟิลด์ class "ชื่ออ้างอิงของตัว Application Server" DATA ข้อมูลที่จะ Append ลงในฟิลด์ General ในรูปของแฟ้มที่ Application Sever จะเรียกใช้ได้)

สำหรับการดึงข้อมูลเข้าฟิลด์แบบ General

2. MODIFY GENERAL ชื่อฟิลด์ General

สำหรับแก้ไขข้อมูลฟิลด์แบบ General

เราจะมาเพิ่มโค้ดในส่วน Activate ตามลิสติงที่ 1